

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: Yasuyuki Nozuyama

Serial No. Not yet assigned

Group Art Unit: Not yet assigned

Filed: October 2, 2003

Examiner: Not yet assigned

Title: **APPARATUS FOR SELECTING TEST PATTERNS FOR LOGIC  
CIRCUIT, COMPUTER IMPLEMENTED METHOD FOR SELECTING  
TEST PATTERNS, AND COMPUTER PROGRAM PRODUCT FOR  
CONTROLLING A COMPUTER SYSTEM SO AS TO SELECT TEST  
PATTERNS**

EXPRESS MAIL NUMBER: EV 301144168 US

DATE OF DEPOSIT: October 2, 2003

I hereby certify that this paper is being deposited with the United States Postal Service "EXPRESS MAIL Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to: Mail Stop PATENT APPLICATION, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313.

  
\_\_\_\_\_  
Susan Pingue

\* \* \*

REQUEST FOR PRIORITY UNDER 35 U.S.C. 119  
AND THE INTERNATIONAL CONVENTION

Assistant Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313

Sir:

In the matter of the above-identified application for patent, notice is hereby given that the applicants claim as priority:

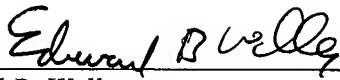
<u>COUNTRY</u>	<u>APPLICATION NO.</u>	<u>MONTH/DAY/YEAR</u>
Japan	P2002-291442	October 3, 2002

A Certified copy of the corresponding Convention Application is being submitted herewith.

Respectfully submitted,

GRAY CARY WARE & FREIDENRICH LLP

Dated: October 2, 2003

By 

*Edward B. Weller*

Reg. No. 37,468

Attorney for Applicant

GRAY CARY WARE & FREIDENRICH  
2000 University Avenue  
East Palo Alto, CA 94303  
Telephone: (650) 833-2436  
Facsimile: (650) 833-2001

## **JAPAN PATENT OFFICE**

This is to certify that the annexed is a true copy of the following application  
as filed with this Office:

Date of Application: October 3, 2002

Application Number: P2002-291442  
[ST.10/C]: [JP2002-291442]

Applicant(s): KABUSHIKI KAISHA TOSHIBA

August 12, 2003

Commissioner:  
Japan Patent Office Yasuo IMAI

Number of Certificate: 2003-3064649

S-927

日本国特許庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日 2002年10月 3日  
Date of Application:

出願番号 特願2002-291442  
Application Number:  
[ST. 10/C]: [JP 2002-291442]

出願人 株式会社東芝  
Applicant(s):

2003年 8月12日

特許庁長官  
Commissioner,  
Japan Patent Office

今井康夫



出証番号 出証特2003-3064649

【書類名】 特許願

【整理番号】 46B019587

【提出日】 平成14年10月 3日

【あて先】 特許庁長官殿

【国際特許分類】 H01L 21/66  
G01R 31/66

【発明の名称】 テストパターン選択装置、テストパターン選択方法、及びテストパターン選択プログラム

【請求項の数】 14

【発明者】

【住所又は居所】 神奈川県川崎市幸区小向東芝町 1 番地 株式会社東芝  
マイクロエレクトロニクスセンター内

【氏名】 野津山 泰幸

【特許出願人】

【識別番号】 000003078

【氏名又は名称】 株式会社 東芝

【代理人】

【識別番号】 100083806

【弁理士】

【氏名又は名称】 三好 秀和

【電話番号】 03-3504-3075

【選任した代理人】

【識別番号】 100068342

【弁理士】

【氏名又は名称】 三好 保男

【選任した代理人】

【識別番号】 100100712

【弁理士】

【氏名又は名称】 岩▲崎▼ 幸邦

## 【選任した代理人】

【識別番号】 100100929

【弁理士】

【氏名又は名称】 川又 澄雄

## 【選任した代理人】

【識別番号】 100108707

【弁理士】

【氏名又は名称】 中村 友之

## 【選任した代理人】

【識別番号】 100095500

【弁理士】

【氏名又は名称】 伊藤 正和

## 【選任した代理人】

【識別番号】 100101247

【弁理士】

【氏名又は名称】 高橋 俊一

## 【選任した代理人】

【識別番号】 100098327

【弁理士】

【氏名又は名称】 高松 俊雄

## 【手数料の表示】

【予納台帳番号】 001982

【納付金額】 21,000円

## 【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【ブルーフの要否】 要

【書類名】 明細書

【発明の名称】 テストパターン選択装置、テストパターン選択方法、及びテストパターン選択プログラム

【特許請求の範囲】

【請求項 1】 論理回路の動作を検証すると共に前記論理回路内に仮定される故障を検出する為のテストパターンの集合から、機能検証カバレッジまたは前記故障の検出率等を基準にして、一部の前記テストパターンを選択する第 1 のテストパターン選択部と、

選択された前記テストパターン（以後、「選択テストパターン」という）を用いて前記故障を検出できるか否かを計算する第 1 の故障シミュレーションを実行する故障シミュレーション部と、

少なくとも前記選択テストパターンによって検出されなかった故障（以後、「第 1 の未検出故障」という）に、前記論理回路のレイアウトに関する重みを付す重み付け部と、

選択されなかった前記テストパターン（以後、「非選択テストパターン」という）の任意の 1 本を用いて、前記第 1 の未検出故障を検出できるか否かを計算する第 2 の故障シミュレーションを実行する前記故障シミュレーション部と、

前記非選択テストパターンの中から、前記第 1 の未検出故障に付された前記重みを考慮して、一部の前記テストパターンを選択する第 2 のテストパターン選択部と

を有することを特徴とするテストパターン選択装置。

【請求項 2】 前記第 1 の故障シミュレーションの結果、故障の検出率向上が所定の基準以下だった選択パターンは、非選択パターンに加えるか、以降の選択から除外することを特徴とする請求項 1 記載のテストパターン選択装置。

【請求項 3】 前記第 1 の未検出故障のリストから、一部の前記故障を抽出する故障抽出部を更に有し、

前記故障シミュレーション部は、前記非選択テストパターンの任意の 1 本を用いて、抽出された前記故障（以後、「第 2 の未検出故障」という）を検出できるか否かを計算し、

前記第 2 のテストパターン選択部は、前記非選択テストパターンの中から、前記第 2 の未検出故障に付された前記重みを考慮して、一部の前記テストパターンを選択する

ことを特徴とする請求項 1 記載のテストパターン選択装置。

【請求項 4】 前記第 1 のテストパターン選択部は、前記テストパターンの機能検証カバレッジを評価する機能検証カバレッジ評価部と、

前記テストパターンの集合から、前記機能検証カバレッジを基準にして、一部の前記テストパターンを選択する選択実行部と

を有することを特徴とする請求項 1 又は 2 記載のパターン選択装置。

【請求項 5】 前記第 1 の故障シミュレーションは、選択テストパターンの任意の 1 本について実行し、前記第 2 のテストパターン選択装置を用いて故障の検出率または検出に伴うレイアウト重みの向上に寄与の小さいものは非選択テストパターンに加えるか以降の選択から除外することを特徴とする請求項 1 又は 2 記載のパターン選択装置。

【請求項 6】 前記故障抽出部は、前記重みを基準にして、前記第 1 の未検出故障のリストから一部の前記故障を抽出することを特徴とする請求項 2 記載のテストパターン選択装置。

【請求項 7】 論理回路の動作を検証すると共に前記論理回路内に仮定される故障を検出する為のテストパターンの集合から、機能検証カバレッジまたは前記故障の検出率等を基準にして、一部の前記テストパターンを選択し、

選択された前記テストパターン（以後、「選択テストパターン」という）を用いて前記故障を検出できるか否かを計算し、

少なくとも前記選択テストパターンによって検出されなかった故障（以後、「第 1 の未検出故障」という）に、前記論理回路のレイアウトに関する重みを付し、

選択されなかった前記テストパターン（以後、「非選択テストパターン」という）の任意の 1 本を用いて、前記第 1 の未検出故障を検出できるか否かを計算し、



前記非選択テストパターンの中から、前記第 1 の未検出故障に付された前記重みを考慮して、一部の前記テストパターンを選択することを特徴とするテストパターン選択方法。

【請求項 8】 前記第 1 の未検出故障のリストから、一部の前記故障を抽出し、

前記非選択テストパターンの任意の 1 本を用いて、抽出された前記故障（以後、「第 2 の未検出故障」という）を検出できるか否かを計算し、

前記非選択テストパターンの中から、前記第 2 の未検出故障に付された前記重みを考慮して、一部の前記テストパターンを選択する

ことを特徴とする請求項 7 記載のテストパターン選択方法。

【請求項 9】 論理回路の動作を検証すると共に前記論理回路内に仮定される故障を検出する為のテストパターンの集合から、機能検証カバレッジまたは前記故障の検出率等を基準にして、一部の前記テストパターンを選択することは、

前記テストパターンの機能検証カバレッジを評価し、

前記テストパターンの集合から、前記機能検証カバレッジを基準にして、一部の前記テストパターンを選択することである

ことを特徴とする請求項 7 又は 8 記載のパターン選択方法。

【請求項 10】 前記第 1 の未検出故障のリストから、一部の前記故障を抽出することは、前記重みを基準にして行うことを特徴とする請求項 8 記載のテストパターン選択方法。

【請求項 11】 コンピュータに、

論理回路の動作を検証すると共に前記論理回路内に仮定される故障を検出する為のテストパターンの集合から、機能検証カバレッジまたは前記故障の検出率等を基準にして、一部の前記テストパターンを選択し、

選択された前記テストパターン（以後、「選択テストパターン」という）を用いて前記故障を検出できるか否かを計算し、

少なくとも前記選択テストパターンによって検出されなかった故障（以後、「第 1 の未検出故障」という）に、前記論理回路のレイアウトに関する重みを付し、

選択されなかった前記テストパターン（以後、「非選択テストパターン」という）の任意の 1 本を用いて、前記第 1 の未検出故障を検出できるか否かを計算し、

前記非選択テストパターンの中から、前記第 1 の未検出故障に付された前記重みを考慮して、一部の前記テストパターンを選択する

ことを実行させることを特徴とするテストパターン選択プログラム。

【請求項 12】 前記第 1 の未検出故障のリストから、一部の前記故障を抽出し、

前記非選択テストパターンの任意の 1 本を用いて、抽出された前記故障（以後、「第 2 の未検出故障」という）を検出できるか否かを計算し、

前記非選択テストパターンの中から、前記第 2 の未検出故障に付された前記重みを考慮して、一部の前記テストパターンを選択する

ことを特徴とする請求項 11 記載のテストパターン選択プログラム。

【請求項 13】 論理回路の動作を検証すると共に前記論理回路内に仮定される故障を検出する為のテストパターンの集合から、機能検証カバレッジまたは前記故障の検出率等を基準にして、一部の前記テストパターンを選択することは、

前記テストパターンの機能検証カバレッジを評価し、

前記テストパターンの集合から、前記機能検証カバレッジを基準にして、一部の前記テストパターンを選択することである

ことを特徴とする請求項 11 又は 12 記載のパターン選択プログラム。

【請求項 14】 前記第 1 の未検出故障のリストから、一部の前記未検出故障を抽出することは、前記重みを基準にして行うことを特徴とする請求項 12 記載のテストパターン選択プログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、テストパターン選択装置、テストパターン選択方法、及びテストパターン選択プログラムに関し、特に、製造品質との相関の強いレイアウト情報を反映した故障検出率を活用したテストパターンの効果的な削減に使用される、L

S I の故障シミュレーションの応用技術に関する。

### 【0002】

#### 【従来の技術】

通常、製造された L S I のチップに対しては、出荷前にそのチップが不良品でないことを確認し、不良品と判定された場合には廃棄する（出荷しない）出荷テスト（製造品質テスト）が行われる。この出荷テストは、論理回路内に存在している可能性のある故障を検出する為のテストパターンを用いて行われる。この出荷テストにおいては、使用するテストパターンが L S I 内の故障をどれだけ確実に検出できるか（故障検出率）が重要なポイントとなるが、この故障検出率の評価は、出荷テストに使用するテストパターンを用いた故障シミュレーションにより行われる。

### 【0003】

また、テストパターンとして、L S I 開発時の機能検証に使用する複数の検証パターンを用いることも多い。検証パターンを用いた故障シミュレーションは、非常に多くの C P U リソースを必要とする。近年の L S I の大規模化等に伴い、検証パターンのサイズ（ステップ数）が増大している。よって、検証パターンをそのままテストパターンに流用することは困難であり、テストパターンとして使用する検証パターンを大幅に削減することが必須条件となっている。（以後、製造品質のテスト（出荷テスト）で使用する検証パターンは、すべて「テストパターン」と呼ぶ。）。

### 【0004】

従来、一定の故障検出率を維持しつつ、使用するテストパターンを大幅に削減する技術がある（例えば、特許文献 1 参照。）。ここではまず、L S I の R T L（Register Transfer Level）記述に対して、検証パターンがどの程度の機能検証カバレッジを達成しているかを評価し、総ての検証パターンと同じ機能検証カバレッジを達成する最小の検証パターンの集合を第 1 のテストパターンとして選択する。選択された第 1 のテストパターンを用いて故障シミュレーションを行って第 1 の未検出故障リストを抽出し、第 1 の未検出故障の一部を無作為抽出して第 2 の未検出故障リストを作成する。非選択検証パターンを用いて第 2 の未検出故障

リストに対して故障シミュレーションを行い、検証パターン毎の検出／未検出故障リストを作成する。このリストを参照して非選択検証パターンの中から故障検出率向上への寄与が大きい検証パターンを第2のテストパターンとして選択する。こうして、最初に選択した第1のテストパターンと、非選択の検証パターンから選択した第2のテストパターンを合わせ、LSIの出荷テストに用いるテストパターンとする。

#### 【0005】

また、論理回路の論理接続情報と、レイアウト情報と、論理接続ノードに仮定される故障と関連付けて抽出されるレイアウト関連パラメータ情報から、重み付き故障リスト或いは重み付き故障検出率を出力する技術（例えば、特許文献2参照。）もある。

#### 【0006】

##### 【特許文献1】

特開 2001-273160号公報

#### 【0007】

##### 【特許文献2】

特開 2000-276500号公報

#### 【0008】

##### 【発明が解決しようとする課題】

上述したように、第1の従来技術は、機能検証カバレッジをパターン選択のための指標とする機能検証カバレッジ評価ツールと、故障検出率をパターン選択のための指標とする故障シミュレーションの応用ツールとを組合せている。これにより、故障検出率を殆ど低下させることなくテストパターン数（サイズ）を大幅に削減することができていた。

#### 【0009】

しかしながら、ここで仮定されている故障は、LSI内部の接続ノードや基本セルの入出力端子といったように、論理的な意味で定義されたものであるに過ぎず、製造テスト品質（不良品リジェクト率）とより密接に関連しているレイアウト要素の情報が考慮されていなかった。したがって、レイアウト要素の情報を考

慮した場合に比べ、有効性が低いテストパターンを選択する可能性があり、結果としてテストパターン数が多くなってしまうという問題があった。

#### 【0010】

本発明はこのような従来技術の問題点を解決するために成されたものであり、その目的は、少ないテストパターンで一定のテスト品質を得ることができ、テストコストを削減することができるテストパターン選択装置、テストパターン選択方法、及びテストパターン選択プログラムを提供することである。

#### 【0011】

##### 【課題を解決するための手段】

上記目的を達成するため、本発明の第1の特徴は、論理回路内に仮定される故障を検出する為に、検証パターンの集合から機能検証カバレッジまたは故障の検出率を基準にして一部の検証パターンを第1のテストパターンとして選択する第1のテストパターン選択部と、選択された第1のテストパターン（選択テストパターン）を用いて故障を検出できるか否かを計算する第1の故障シミュレーション部と、第1のテストパターンによって検出されなかった故障（第1の未検出故障）に論理回路のレイアウトに関する重みを付す重み付け部と、選択されなかった検証パターン（非選択検証パターン）を用いて第1の未検出故障を検出できるか否かを計算する第2の故障シミュレーション部と、非選択検証パターンの中から第1の未検出故障に付された重みを考慮して一部の検証パターンを第2のテストパターンとして選択する第2のテストパターン選択部とを有するテストパターン選択装置であることである。論理回路の出荷テストは、第1及び第2のテストパターンを用いて行われる。

#### 【0012】

本発明の第1の特徴によれば、第1のテストパターン選択部が選択する第1のテストパターンには漏れたが、レイアウトに関する重みを考慮した故障検出率の向上に大きな寄与があり、かつテストパターン長の増加が少ない検証パターンを第2のテストパターンとして優先的に選択することができる。

#### 【0013】

本発明の第2の特徴は、論理回路内に仮定される故障を検出する為に、検証パ

ターンの集合から機能検証カバレジまたは故障の検出率を基準にして一部の検証パターンを第1のテストパターンとして選択し、選択された第1のテストパターン（選択テストパターン）を用いて故障を検出できるか否かを計算し、第1のテストパターンによって検出されなかった故障（第1の未検出故障）に論理回路のレイアウトに関する重みを付し、選択されなかった検証パターン（非選択検証パターン）を用いて第1の未検出故障を検出できるか否かを計算し、非選択検証パターンの中から第1の未検出故障に付された重みを考慮して一部の検証パターンを第2のテストパターンとして選択するテストパターン選択方法であることである。論理回路の出荷テストには、第1及び第2のテストパターンを用いる。

#### 【0014】

本発明の第3の特徴は、コンピュータに、論理回路内に仮定される故障を検出する為に、検証パターンの集合から機能検証カバレジまたは故障の検出率を基準にして一部の検証パターンを第1のテストパターンとして選択し、選択された第1のテストパターン（選択テストパターン）を用いて故障を検出できるか否かを計算し、第1のテストパターンによって検出されなかった故障（第1の未検出故障）に論理回路のレイアウトに関する重みを付し、選択されなかった検証パターン（非選択検証パターン）を用いて第1の未検出故障を検出できるか否かを計算し、非選択検証パターンの中から第1の未検出故障に付された重みを考慮して一部の検証パターンを第2のテストパターンとして選択することを実行させるテストパターン選択プログラムであることである。

#### 【0015】

本発明の第2及び第3の特徴によれば、第1のテストパターン選択には漏れたが、レイアウトに関する重みを考慮した故障検出率の向上に大きな寄与があり、かつパターン長の増加が少ない検証パターンを第2のテストパターンとして優先的に選択することができる。

#### 【0016】

##### 【発明の実施の形態】

以下図面を参照して、本発明の実施の形態を説明する。図面の記載において同一あるいは類似部分には同一あるいは類似な符号を付している。

## 【0017】

図1に示すように、本発明の実施の形態に係るテストパターン選択装置は、実施の形態に係るパターン選択方法を実行する為の機能手段を備えた演算部1と、データや命令を転送する為のバス11を介して演算部1に接続されたデータ記憶部及びプログラム記憶部12とを少なくとも有する。演算部1は、第1のテストパターン選択部20と、故障シミュレーション部21と、重み付け部23と、故障抽出部24と、第2のテストパターン選択部26とを有する。データ記憶部は、故障リスト記憶部2と、テストパターン記憶部3と、レイアウト情報記憶部4と、レイアウト関連パラメータ情報記憶部4.5と、回路情報記憶部5と、基本セルライブラリ記憶部6と、選択情報記憶部7とを有する。レイアウト情報記憶部4にはレイアウト情報36、レイアウト関連パラメータ情報記憶部4.5にはレイアウト関連パラメータ情報36が格納されている。回路情報記憶部5にはRTLネット37及びゲートレベルネット38の情報が格納されている。

## 【0018】

演算部1は、通常のコンピュータシステムの中央処理装置(CPU)の一部として構成すればよい。第1のテストパターン選択部20、故障シミュレーション部21、重み付け部23、故障抽出部24、及び第2のテストパターン選択部26は、それぞれ専用のハードウェアで構成しても良く、通常のコンピュータシステムのCPUを用いて、ソフトウェアで実質的に等価な機能を有する機能手段として構成しても構わない。

## 【0019】

故障リスト記憶部2、テストパターン記憶部3、レイアウト情報記憶部4、レイアウト関連パラメータ情報記憶部4.5、回路情報記憶部5、基本セルライブラリ記憶部6、選択情報記憶部7、及びプログラム記憶部12は、それぞれ、半導体ROM、半導体RAM等の半導体メモリ装置、磁気ディスク装置、磁気ドラム装置、磁気テープ装置などの外部記憶装置で構成してもよく、CPUの内部の主記憶装置で構成しても構わない。

## 【0020】

また、バス11には、入出力制御部8を介して、操作者からのデータや命令な

どの入力を受け付ける入力装置 9 と、テストパターンの選択結果を出力する出力装置 10 が接続されている。入力装置 9 は、キーボード、マウス、ライトペンまたはフレキシブルディスク装置などで構成されている。また出力装置 10 は、ディスプレイ装置やプリンタ装置などにより構成されている。

#### 【0021】

図 1 に示した演算部 1 で実行される各処理の入力データは、データ記憶部に記憶され、プログラム命令はプログラム記憶部 12 に記憶されている。そしてこれらの入力データやプログラム命令は必要に応じて CPU に読み込まれ、CPU の内部の演算部 1 によって、演算処理が実行されるとともに、一連の演算処理の各段階で発生した数値情報などのデータは、RAM や磁気ディスクなどのデータ記憶部に格納される。

#### 【0022】

第 1 のテストパターン選択部 20 は、論理回路内に仮定される故障を検出する為に、検証パターンの集合から、機能検証カバレッジまたは故障の検出率を基準にして、一部の検証パターンを第 1 のテストパターンとして選択する。第 1 のテストパターン選択部 20 により選択されたテストパターンは、選択テストパターン 33 としてテストパターン記憶部 3 に記憶される。一方、第 1 のテストパターン選択部 20 により選択されなかった検証パターンは、非選択検証パターン 34 としてテストパターン記憶部 3 に記憶される。ここで、「論理回路」は、LSI 全体の論理回路或いは LSI の一部の機能ブロックなどの論理回路を示す。「機能検証カバレッジ」（「RTL コードカバレッジ」とも言う）は、LSI の論理回路の RTL 記述に対し、一般に多数本から構成される検証パターンがどの程度十分に検証を行っているかを確認するための指標であり、RTL 記述の実行可能な文をどの程度実行したかというステートメントカバレッジ (statement coverage)、分岐文の真と偽をどの程度実行したかというブランチカバレッジ (branch coverage)、分岐の論理式の各信号の組み合わせをどの程度実行したかというコンディションカバレッジ (condition coverage)、各信号がどの程度変化したかというトグルカバレッジ (toggle coverage)、等がある。「故障の検出率」は、論理回路内に仮定される故障に対してテストパターンを用いて故障シミュレーションを実施



した場合におけるテストパターンが故障を検出する程度を示す。

### 【0023】

具体的には、機能検証カバレッジを基準にする場合、第1のテストパターン選択部20は、検証パターンの機能検証カバレッジを評価する機能検証カバレッジ評価部28と、検証パターンの集合から、機能検証カバレッジを基準にして、一部の検証パターンを選択する選択実行部29とを有する。近年、LSIのRTL記述に対し、一般に多数本の個別検証パターンで構成される検証パターンの集合がどの程度の機能検証カバレッジ（「RTLコードカバレッジ」ともいう）を達成しているかを調べる機能検証カバレッジ評価ツールが市販されている。さらに、検証パターン毎に得られる機能検証カバレッジの評価結果を利用して、全ての検証パターンと同じ機能検証カバレッジを達成する最小本数の検証パターンの組を選択する検証パターン選択ツールが市販されている。機能検証カバレッジ評価部28として機能検証カバレッジ評価ツールを使用し、選択実行部29として検証パターン選択ツールを使用することができる。

### 【0024】

機能検証カバレッジ評価部28及び選択実行部29においては、テストパターンとして、LSI開発時の機能検証に使用する検証パターンを適宜編集或いは流用して使用している。LSIの中でも特にCPUのような論理回路において、この検証パターンは、技術者にとって必要な検証項目をカバーしたことが理解しやすいように開発・作成され、検証パターンの長さを最小にしようという意図のもとに作成される訳ではない。したがって、機能検証カバレッジ評価ツール及びテストパターン選択ツールを用いた場合、選択テストパターン33のサイズ（ステップ数）は、全検証パターンの数分の1～1/10未満になることもしばしばある。

### 【0025】

選択テストパターン33の集合は、機能検証カバレッジという面ではテストパターン記憶部3内の全テストパターン（検証パターン）の集合と等価となる。したがって、RTLネット37に対し機能検証カバレッジが100%に近い値に到達しているという条件下では、ゲートレベルネット38に対しても同様に高い故障の検出率を達成できると期待される。ここで、ゲートレベルネット38とは、論理

回路の内部の多数の基本セル及び入出力端子、及びこれらの間を接続する配線からなるゲートレベルの論理接続ネットを示す。基本セルは、組合せ論理ゲート（簡単な素子も含む）と、ラッチ、フリップフロップ等の記憶セルによって構成されている。さらに、広い意味でROM、RAMといったメモリやその他特殊な回路の動作モデル（故障シミュレーション用）を含む場合もある。

#### 【0026】

第1の故障シミュレーション部21は、選択テストパターン33を用いて故障を検出できるか否かを計算する。ここで、「故障」には、故障リスト記憶部2に記憶されている故障リストが含まれる。また、故障リスト記憶部2に記憶されている故障リストは、通常、第1の故障シミュレーション部21によってゲートレベルネット38に対応するものが作成される。故障リストは故障シミュレーション実行前に一旦故障リスト部2に格納しても良いし、故障シミュレーション実行後に故障リスト部2に格納するようにしても良い。本実施例では前者の場合について説明する。「故障」のモデルとしては、ゲートレベルネット38内の任意の1個の接続ノード（ネット）の値が0または1に固定されたと仮定する単一ネット縮退故障モデルが含まれ、また、LSI内部の基本セルの入出力端子部分に0／1縮退故障を仮定する単一ピン縮退故障モデルが更に含まれ、「故障」はいずれかのモデルに従って仮定される。仮定される故障の最大数（総ての可能な箇所に故障仮定した場合の故障数）は、それぞれのモデルにおいて、 $2 \times$ （接続ノード数）個、 $2 \times$ （入出力端子数＋基本セルの入出力端子数）個となる。ここでは、故障シミュレーションは、ゲートレベルネット38を対象とする場合について説明するが、ゲートレベルネット38の中に素子レベルの接続ネットが混在しても構わない。

#### 【0027】

具体的には、故障シミュレーション部21は、故障リスト記憶部2に記憶されている故障リスト30aに対して、選択テストパターン33と、ゲートレベルネット38と、基本セルライブラリ6とを用いて第1の故障シミュレーションを実行する。実行結果は、選択テストパターン33によって故障リストの故障が検出されたかまたは検出されなかったかという情報を含んだ第1の（未）検出故障リ

スト 3 0 a として故障リスト記憶部 2 に上書き・記憶される。故障検出率はこのリスト 3 0 a から算出される。通常は故障シミュレーション部 2 1 が算出機能を有してゐる。

#### 【 0 0 2 8 】

なお、詳細に見ると、故障シミュレーションの結果、故障リスト記憶部 2 に記憶されている故障は、選択テストパターン 3 3 によって確実に検出された (hard detected) 故障、ポテンシャル検出 (potentially detected) 故障、ハイパーアクティブ (hyperactive) 故障、発振 (oscillatory) 故障、冗長な故障等に分類される (冗長な故障は故障リスト作成時に分類がなされることもある)。ここで、ポテンシャル検出故障とは、例えば正常な場合には 0 または 1 の確定出力であるものが、その故障の存在の結果不定値 (x) になるといった範疇の故障を示す。発振故障は故障の存在により回路が発振してしまう故障である。また、ハイパーアクティブ故障とは、特定のテストパターンでの故障シミュレーションにおいて CPU 時間の無意味な増加を招く恐れがあるため計算を中断した故障を示す。故障シミュレーションを複数回実施してポテンシャル検出が所定の回数を越えた場合、ポテンシャル検出故障は、ほぼ確実に検出されたと見做してポッシブル検出 (possibly detected) と分類される場合もある。これらの分類の中で第 1 の (未) 検出故障リスト 3 0 a に本発明における未検出故障として含まれるものは、基本的には、完全な未検出故障と、ハイパーアクティブ故障とである。或いは、発振故障、ポテンシャル検出故障の一部 (割合) まで含ませても構わない。さらに、最近では、複数回の確実な検出が L S I 内部の実際の不良の検出能力の向上に有効であることが分かってきており、所定の検出回数に満たない「確実な検出」故障まで含ませても良い。なお、本発明の上記「未検出」扱いの故障を、故障リスト 3 0 a と異なる故障リストにまとめるようにしても良い。

#### 【 0 0 2 9 】

重み付け部 2 3 は、第 1 の (未) 検出故障リスト 3 0 a の故障 (少なくとも前記「未検出」扱いの故障) に、論理回路のレイアウトに関する重みを付す。論理回路のレイアウトに関する重みは、レイアウト情報記憶部 4 に格納された対象とする L S I のレイアウト情報 3 6 から、レイアウト関連パラメータ情報記憶部 4

． 5 に格納されたレイアウト関連パラメータ情報 3 6 . 5 に従って抽出される。レイアウト情報としては、自動配置配線ツールが L S I の基本セルの配置及び基本セルの入出力端子（L S I の入出力端子含む）間の配線処理を行った結果の情報や、さらには、基本セル内部のレイアウト情報まで含んだものもある。レイアウト関連パラメータ情報 3 6 . 5 は、対象とする L S I のレイアウト情報から着目する論理接続ノード又は入出力端子に仮定される故障と関連付けて抽出されるべき情報である。重み付け部 2 3 は、レイアウト情報 3 6 から、レイアウト関連パラメータ情報 3 6 . 5 に従って、L S I 内の論理接続ノード又は入出力端子に対応する所定のレイアウト要素の量を重みとして抽出（算出）し、関連する故障に対応付ける。なお、所定のレイアウト要素の量を、論理回路のレイアウトに関する重みとする代わりに、操作者が外部から論理回路のレイアウトに関する重みを直接指示しても構わない。

#### 【 0 0 3 0 】

具体的には、レイアウト関連パラメータ情報 3 6 . 5 には、故障発生の原因になりやすく、論理接続ノード又は入出力端子に仮定された故障と対応付けしたいレイアウト要素、その組合せ、乃至それらを決定する為の制約条件・規則などに関する情報が含まれる。故障と対応付けしたいレイアウト要素には、例えば、隣り合う配線との距離が最小ピッチとなっている配線部分の長さや、最小サイズのコンタクトだけで接合された部分の個数などが含まれる。

#### 【 0 0 3 1 】

故障の重み付けにより、論理接続ノード又は入出力端子に仮定された故障とレイアウト上現実に故障が発生し易い個所との対応を明確にすることができる。重みが付された第 1 の（未）検出故障リスト 3 0 a は、第 1 の（未）検出故障リスト（重み付き） 3 0 b として故障リスト記憶部 2 に記憶される。重みを付す前の故障リストと、重みを付した後の故障リストは、例えば表 1 及び表 2 に示すようになる。

#### 【 0 0 3 2 】

【表 1】

重みなし	
故障A	D
故障B	U
故障C	D



重みあり			
故障A	D	$M1\_312+M2\_57+M1\_2199$	WA
故障B	U	$M1\_15+M2\_1187+M1\_745$	WB
故障C	D	$M1\_2935+M2\_583+M1\_1865$	WC

【表 2】

重みなし	
故障A	D
故障B	U
故障C	D



重みあり						
故障A	D	W1A	$V12\_31+V21\_1782$	W2A	$M1\_3158+M2\_186+M1\_1811$	WA
故障B	U	W1B	$V12\_785+V21\_5581$	W2B	$M1\_437+M2\_2593+M1\_761$	WB
故障C	D	W1C	$V12\_3957+V21\_833$	W2C	$M1\_2814+M2\_58+M1\_1527$	WC

表 1 及び表 2 中の“D”は故障シミュレーションにより確実に検出されたことを示し、“U”は検出されなかったことを示す。縮退故障である“故障 A”は、例えば L S I 内部の接続ノード名と、“0”への縮退(stuck-at 0 : S A 0)或いは“1”への縮退(stuck-at 1 : S A 1)を区別する情報(S A 0 / S A 1)とによって構成されている。さらに、代表故障(representative fault)か等価

故障 (equivalent fault) かを区別する情報 (例えば rep/eq) が追加されることがある。故障の中には、LSI 外部から総ての可能なテストパターンを印加しても、誤り出力 (正常回路の場合と異なる出力) が互いに全く同じとなって区別ができないものがあり、そうした故障同士を等価故障という (こうした等価故障の組は多数存在する)。故障シミュレーションでは、CPU 時間制約の為、等価故障の各組毎に各 1 個を代表故障として選び、実行するのが一般的である。故障リストでは、代表故障のすぐ下の行以降に関連する等価故障を記載することが多い。未検出故障の正しい優先度付け (テストパターン追加の優先度付け) のためには、この等価故障まで考慮して重みを算出できるようにしておくことが望ましい。ここで、接続ノード名としては、通常 “/I15/Ixyz/A[12]” のように、接続ネットの階層を反映した形で記載される。また、表 1 では、各故障 A, B, C に対し、それぞれ重み  $W_A$ ,  $W_B$ ,  $W_C$  が付されている。表 2 では、各故障 (A, B, C) に対し、付加すべき複数 (2 種類) のレイアウト要素の重み  $w_1$  (異なるメタル配線間の最小接続コンタクト数)、 $w_2$  (メタル配線長) の重みが記載され、それぞれに対応する具体的なレイアウト要素の詳細 (例えば  $w_1 A$  に対しては個々の識別名の和 ( $V12\_31 + V21\_1782$ ) の形で表記) が付記されており、さらに最終の重み  $W$  が付された形となっている。例えば、故障 A については、 $W_A = x \cdot w_1 A + y \cdot w_2 A$  という式によって算出される ( $x$ ,  $y$  は適当な定数: 故障 B, C についても同様)。この計算は重み付け部 23 によって実行される。こうした式の情報は、予め必要なレイアウト関連パラメータ情報を検討して各故障に付加すべき重みを決定し、重み付け部 23 に計算プログラムとして格納しておくが良い。この際、定数部分については、レイアウト関連パラメータ 36.5 の一部とパラメータ設定できるようにしても良い。なお、各故障に重みを付すに当たって注意すべき点は、複数の故障が特定のレイアウト要素を共有する場合がしばしばあることである。この例を図 2 に示す。基本セル C0 から C1, C2, C3 に配線が接続されており (接続ノード A)、それぞれの端子 Z, A1, A2, A3 に故障 flt0, flt1, flt2, flt3 (総て 0 縮退か、または 1 縮退故障) が定義されている。ノード A を構成するレイアウト要素は、第 1 層メタル配線 M1\_0 ~ M1\_3、第 1 層・第 2 層メタル間コンタクト (VIA) (最小サ

イズのもの) V12\_0~V12\_3、第2層メタル配線M2\_1~M2\_3 (実際は1本の配線、計算の便宜上) である。ここで、故障flt0、flt2が、検出(D)、flt1、flt3が未検出(U) とすると、上記レイアウト要素の内、M1\_0、V12\_0、M2\_2、V12\_2、M1\_2の部分に生じた不良は検出されることになるが、その他のレイアウト要素に対応した部分に発生した不良は検出できない。ここで未検出の故障flt1に対応するレイアウト要素について見てみると、M1\_0、V12\_0、M2\_2、M2\_1、V12\_1、M1\_1であるが、flt0、flt2の検出によって、M1\_0、V12\_0、M2\_2、M1\_2に発生する不良は確実に検出できると判明しているため、flt0、flt2が検出という条件下で未検出故障flt1に対応するレイアウト要素は、M2\_1、V12\_1、M1\_1となる。また、flt0、flt3が検出で、flt1、flt2が未検出の場合、flt1に対応するレイアウト要素は、M2\_2、M2\_1、V12\_1、M1\_1となる。このように、特定のレイアウト要素が複数の故障に共有される場合があるため、未検出故障に対応させるべきレイアウト要素は、他の故障の検出(未検出)状況に依存する場合があることに注意し、重み付けする必要がある。

### 【0033】

故障抽出部24は、第1の(未)検出故障リスト(重み付き)30bから、レイアウトに関する重みを基準にして、本発明において「未検出」扱いとする故障の一部を抽出し、「未検出」故障リスト31を作成する。即ち、故障抽出部24は、レイアウトに関する重みに対応(基本的には比例)した確率で、一部の未検出扱いの故障の一部を無作為抽出する(総てを抽出しても良い)。これにより、意味のある(実際発生する不良と強い相関のある)故障を用いてテストパターンの選択を実施することができる。なお、故障抽出部24は、レイアウトに関する重みを基準にする代わりに、第1の(未)検出故障リスト(重み付き)30b内の「未検出」扱いの故障を重みを考慮せずに無作為抽出(ランダム・ランプリング)しても構わない。

### 【0034】

故障抽出部24は、故障シミュレーションの実行に必要なCPU時間を減少させることを目的とする。故障シミュレーションの実行時間は、抽出の割合にほぼ

比例する。抽出の割合を低くすると、抽出されなかった故障の検出に関する推定の信頼度が低下し、また、追加選択されるテストパターンが減り、故障の検出率の信頼性が低下することになる。しかし、前述したように、機能検証カバレッジ評価部 28 及び選択実行部 29 が選択したテストパターン 33 は、故障の検出率（故障シミュレーション部 21 が故障シミュレーション実行後の故障リスト 30b に対応する結果として出力）が全テストパターンによるものになりに近くなると考えられる。したがって、抽出の割合をある程度低くしても、故障検出率の結果に大きな食い違いが出る余地は小さくなると考えられる。しかしながら、抽出の割合をあまり低くしすぎると、テストパターン選択の信頼性が低下するため、経験的に適当な値にすることが望ましい。

#### 【0035】

次に、故障シミュレーション部 21 は、非選択テストパターン 34 を用いて、第 2 の「未検出」故障リスト 31 の故障を検出できるか否かを計算する。即ち、故障シミュレーション部 21 は、第 2 の「未検出」故障リスト 31 に対して、非選択テストパターン 34 と、ゲートレベルネット 38 と、基本セルライブラリ 6 とを用いて第 2 の故障シミュレーションを実行する。故障シミュレーションは、非選択テストパターン 34 の任意の 1 本について実行される。非選択テストパターン 34 によって検出された故障は、追加検出故障リスト 32 として故障リスト記憶部 2 に記憶される。追加検出故障リスト 32 は、総ての非選択テストパターン 34 についてテストパターン毎に作成される。

#### 【0036】

なお、第 2 の「未検出」故障リスト 31 の故障には重みが付されている。故障シミュレーションは、通常、重み付きの故障に対して実行できない。しかし、簡単なプログラムにより、故障シミュレーション部 21 への入力にはレイアウト要素の重みを省略した故障リストを用いるようにすることが可能である。また、故障シミュレーション部 21 から出力される追加検出故障リスト 32 には再び重みが付される。

#### 【0037】

第 2 のテストパターン選択部 26 は、非選択テストパターン 34 の中から、追



加検出故障リスト 32 の故障に付された重みを考慮した故障の検出率を基準にして、一部のテストパターンを選択する。このとき、第 2 のテストパターン選択部 26 は、重み付きの「未検出」故障リスト 31、重み付きの追加検出故障リスト 32 及び選択情報記憶部 7 に記憶されている選択情報を入力情報として用いる。ここで、「選択情報」は、故障リストから直接には得られない、パターン選択のために必要な情報を示す。第 2 のテストパターン選択部 26 によって選択されたテストパターンは、追加選択テストパターン 35 としてテストパターン記憶部 3 に記憶される。追加選択テストパターン 35 及び選択テストパターン 33 からなるテストパターンの集合は、テストパターン記憶部 3 内の全テストパターンで到達することができる故障の検出率とほぼ同じ故障の検出率を達成することができる。第 2 のテストパターン選択部 26 の詳細な構成は図 3 を参照して後述する。

#### 【0038】

最後に、故障シミュレーション部 21 は、追加選択テストパターン 35 を用いて、第 1 の（未）検出故障リスト 30b の「未検出」故障を検出できるか否かを計算する。即ち、故障シミュレーション部 21 は、第 1 の（未）検出故障リスト 30b に対して、追加選択テストパターン 35 と、ゲートレベルネット 38 と、基本セルライブラリ 6 とを用いて第 3 の故障シミュレーションを実行する。故障シミュレーション部 21 は、最終的な故障の検出率を確認することを目的とする。最終的な故障の検出率は、追加選択テストパターン 35 及び選択テストパターン 33 からなるテストパターンの集合により達成される故障検出率を示す。なお、故障シミュレーション部 21 は、第 1 の（未）検出故障リスト（重み付き）30b の代わりに、第 1 の（未）検出故障リスト 30a を用いても構わない。

#### 【0039】

故障シミュレーション部 21 による第 3 の故障シミュレーションによって、論理回路（ゲートレベルネット 38）に対する最終的な重み付き故障検出率及び未検出故障のリスト（30c、d）などが得られる。「未検出故障」とは、選択テストパターン 33 及び追加選択テストパターン 35 によって検出されなかった故障であり、基本的には完全な未検出故障とハイパーアクティブ故障からなり、場合によってポテンシャル検出故障及び発振故障の一部（割合として含む）からな

る。第3のシミュレーションにより、故障抽出部24によって抽出されなかった「未検出」扱いの故障についても追加選択テストパターン35による検出／未検出が明確となり、対象とするLSIなどの論理回路の未検出故障解析、追加テストパターンの作成を容易にすることができる。なお、（レイアウト要素の）重み付きの故障検出率は、（1）式により定義される。

#### 【0040】

重み付き故障検出率

$$= (\text{検出された故障の重みの総和}) / (\text{全故障の重みの総和}) \quad \dots (1)$$

図3に示すように、第2のテストパターン選択部26は、追加検出故障数評価部51と、候補テストパターン選択部52と、ベストパターン選択部53と、ベストパターン評価部54と、評価対象リスト変更部55とを有する。

#### 【0041】

追加検出故障数評価部51は、非選択テストパターン34の1本ごとに、評価対象となる「未検出」故障リスト（評価対象リスト）の「未検出」故障と追加検出故障リスト32内の追加検出故障（またはレイアウト要素）を比較し、追加検出故障数（またはその重み）が0（または殆ど0）である、即ち故障を全く（または殆ど全く）追加検出していない非選択テストパターン34を、第2のテストパターン選択部26の選択対象から除外する。但し、最初のステップでは、上記比較は不要であり、追加検出故障リスト32内で追加検出故障数（またはその重み）が0または0に近いものを除外するだけで良い。

#### 【0042】

候補テストパターン選択部52は、（2）式に示す評価値（EV）の条件式を満たす非選択テストパターン34の1本以上を候補テストパターンとして選択する。

#### 【0043】

$$EV(\text{パターン}) \geq \alpha \quad \dots (2)$$

なお、 $\alpha$ は、 $0 < \alpha \leq 1$ である。具体的には、まず、候補テストパターン選択部52は、非選択テストパターン34の1本毎の追加検出故障リスト32（重み

付き)、選択情報記憶部 7 内の選択情報及び評価対象リスト (最初は重み付きの「未検出」故障リスト 31) を元に、非選択テストパターン 34 毎に評価値 (E V) を計算する。そして、その評価値 (E V) が一定の値 ( $\alpha$ ) 以上である 1 又は 2 以上のテストパターンを候補テストパターンとして選択する。評価値 (E V) の計算式の例を (3) 乃至 (5) 式に示す。

#### 【0044】

$$EV(\text{パターン}) = r \times (\text{規格化された追加検出故障重み}) \\ + (1 - r) / (\text{規格化されたテストパターン長}) \quad \dots (3)$$

(規格化された追加検出故障重み)

$$= (\text{その非選択テストパターンで追加検出} \\ \text{された故障の重みの総和}) / \\ (\text{その時点での全ての非選択テストパターン別の追加検出} \\ \text{された全故障重みの総和の最大値}) \quad \dots (4)$$

(規格化されたテストパターン長)

$$= (\text{その非選択テストパターンの長さ (ステップ数)}) / \\ (\text{その時点での全ての非選択テストパターン別の最大長さ (ステ} \\ \text{ップ数)}) \quad \dots (5)$$

ここで、上記の各故障の重みは図 2 で説明したように、検出された故障による影響を正しく考慮しつつ、前出の表 2 のリストを用い、前出の重み付け式に従って算出する。

#### 【0045】

なお、 $r$  は、 $0 \leq r \leq 1$  である。 $r$  は、追加検出故障のレイアウト要素による重みとテストパターン長の逆数 ( $1 / \text{テストパターン長}$ ) の間の重み付けを調整するためのパラメータであり、選択情報として与えられる。 $r$  の値が大きくなると、テストパターン長よりも追加検出故障の重みの方を重視することになり、重み付き故障検出率の向上へ寄与するテストパターンが優先的に選択される。しかし、テストパターン長が長く、テスト時間が長いテストパターンが選択されてし

まう欠点がある。第3の故障シミュレーションのCPU時間が長くなるという影響もある。

#### 【0046】

一方、 $r$ の値が小さくなると、テストパターン長の短いテストパターンが優先的に選択されることになる。しかし、選択したテストパターンの本数が増えても、それに比例して追加検出される故障数が速やかに増加していかない。したがって、所定の故障検出率に到達するまで第2のテストパターン選択部26内においてより多くの計算を実行させる必要が生じ、全体のCPU時間が増加してしまう欠点が出てくる。実際の使用では、経験的に妥当な値を設定する必要がある。

#### 【0047】

ベストパターン選択部53は、候補テストパターンの中で、評価値(EV)が最大であるテストパターンを含め適当な本数をベストパターンとして選択する。

#### 【0048】

ベストパターン選択部53は、まず、評価値(EV)が最大の「最初の」ベストパターン1本の追加検出故障リスト32と、評価対象リストB0とを比較する。ここで、「評価対象リストB0」の初期値は、第2の「未検出」故障リスト31を示す。具体的には、評価対象リストB0の「未検出」故障のうち、ベストパターンによってどの故障(及びレイアウト要素)が検出され、またどの故障(及びレイアウト要素)が未検出のままかを求め、検出されたものについては、評価対象リストB0から除外し、除外後のリストを新たに評価対象リストB0とする。このリストB0と残りの候補テストパターンの追加検出故障リスト32とを比較する。評価対象リストB0に対して残りの各候補テストパターンによって検出される故障に伴う重みが予め決めておいた基準に達していない場合、その候補テストパターンはこの候補テストパターンの組での選択では選択しない。例えば、ベストパターン選択部53は、(6)式の条件を満たさない候補テストパターンは選択しない。

#### 【0049】

$EV(\text{候補パターン}) \geq$

$\beta \times (EV(\text{最初のベストパターン}) - \text{可能な最低の評価値}(0))$

．．．（６）

なお、 $\beta$ は、 $0 < \beta \leq 1$ である。EV（候補パターン）は、評価対象の候補テストパターンの評価値（EV）を示す。EV（最初のベストパターン）は、ベストパターン選択部 53 が全候補テストパターンの中から最初に選択したベストパターンの評価値（EV）を示す。（６）式を満たす候補テストパターン中、評価値（EV）が最大のものが２番目のベストパターンとして選ばれ、評価対象リスト変更部 55 が評価対象リスト B0 の「未検出」故障中、追加検出した故障を除外して新たな評価対象リスト B0 とする。これを順次、（６）式を満たす候補テストパターンがなくなるまで繰り返す。こうして、１本以上の「ベスト」パターンが選択される。

#### 【0050】

上述した仕組みにより、候補テストパターンとして選択された時には追加検出故障数が多かったものの、検出した故障が互いに似通ったテストパターンが複数ある場合に、これらのテストパターンが重複して選択され、冗長なテストパターンが選択される、といったことが生じない。即ち、ベストパターン選択部 53 は、追加選択テストパターン 35 として選択されるテストパターンの数が不用意に増加させるのを防ぐようになっている。

#### 【0051】

なお、テストパターン数を最少にしたい場合、故障抽出部 24 が総ての「未検出」故障を抽出して（100% サンプリング）、第２の「未検出」故障リスト 31 を作成した上で、ベストパターンを１本ずつ選択していけば良い。但し、この場合、CPU 時間が大幅に増加する場合がある。

#### 【0052】

評価対象リスト変更部 55 は、候補テストパターンが（６）式の条件を満たしている場合に、評価対象リスト B0 からこの候補テストパターンの追加検出故障リスト 32 に記載された故障（及びレイアウト要素）を削除して、これを新たな評価対象リスト B0 と定義する。なお、候補テストパターンが（６）式の条件を満たしていない場合には、評価対象リスト B0 は変更されない。

#### 【0053】

1 回の候補テストパターン選択（候補テストパターン選択部 52 による）の度に、総ての候補テストパターンの評価が終了するまで、ベストパターン選択部 53 及び評価対象リスト変更部 55 は繰り返し実施される。次には、更新された評価対象リスト B0 を新たな評価対象リスト A0 とし、選択されなかったテストパターンの中から、候補テストパターン選択部 52 が候補テストパターンを選択し、同様な処理を行う。これを順次繰り返し、最終的に候補テストパターンが選択されなくなるまで（即ち、追加検出故障が 0 となるまで）、または、候補テストパターン選択の有効性が殆どなくなるまで実施することが望ましい。後者のケースは、例えば、ベストパターンの追加選択に比して故障検出率の向上が殆ど見込めなくなるまで、繰り返し実施することが望ましい。繰り返して実施することにより、効率的に未検出故障を検出している幾つかのテストパターンを選び出すことができる。

#### 【0054】

以上説明したように、実施の形態に係るテストパターン選択装置によれば、第 1 のテストパターン選択部 20 が選択するテストパターンには漏れたが、レイアウトに関する重みを考慮した故障検出率の向上に大きな寄与があり、かつパターン長の増加が少ないテストパターンを優先的に選択することができる。

#### 【0055】

次に、図 4 を参照して、本発明の実施の形態に係るテストパターン選択方法について説明する。

#### 【0056】

(イ) まず、S01 段階において、論理回路内に仮定される故障を検出する為のテストパターンの集合から、機能検証カバレッジまたは故障の検出率等を基準にして、一部のテストパターンを選択する。具体的には、まず、テストパターンの機能検証カバレッジを評価する。そして、テストパターンの集合から、機能検証カバレッジを基準にして、一部のテストパターンを選択する。

#### 【0057】

選択されたテストパターン（選択テストパターン）33 の集合は、全テストパターンより大幅にサイズが小さいが、全テストパターンで期待される故障の検出

率に比較的近い故障検出率を達成すると思われる。しかし、機能検証カバレッジは、対象としているLSIのRTL記述レベルの機能検証項目を活性化したということであり、ゲートレベルの記述において、テスト容易性(testability)の内、テスト制御性(controllability)に対してはよい指標となる。しかし、テスト容易性の内、テスト観測性(observability)については確実に反映しているとは言えない。したがって、全テストパターンと同程度の故障検出率を確保するためには、第1次解としては有効だが、故障検出率が全テストパターンで本来得られるであろう故障検出率よりも幾分(数%程度)低くなる可能性が高い。また、未検出故障の調査及びテストパターンの追加作成は、多くの技術者リソースを必要とする困難な作業である。

#### 【0058】

そこで、以下に示す手順を経て、選択されなかったテストパターン(非選択テストパターン)34の中から、未検出故障を効率的に検出するテストパターンを追加選択する。

#### 【0059】

なお、選択テストパターン33を得る為には、他の手段もあり、例えば、対象故障を低いサンプリング率でサンプリングし、全テストパターンで個別に故障シミュレーションし、その結果で検出率の高いものを選択することもできる。最悪の場合、検証担当者が検証項目・内容から推定して選別することも可能である。

#### 【0060】

(ロ) 次に、S02段階において、選択テストパターン33を用いて故障を検出できるか否かを計算する。即ち、故障リストに対して、選択テストパターン33と、ゲートレベルネット38と、基本セルライブラリ6とを用いて故障シミュレーションを実行する。実行結果として出力される第1の(未)検出故障リスト30aは、少なくとも選択テストパターン33によって検出されなかった故障のリストを含む。「未検出」は有効なテストパターンを選別する為に定義される。

#### 【0061】

(ハ) 次に、S03段階において、第1の(未)検出故障リスト30aの故障に、論理回路のレイアウトに関する重みを付す。論理回路のレイアウトに関する

重みは、レイアウト情報 36 からレイアウト関連パラメータ情報 36.5 に従って抽出される。故障の重み付けにより、論理接続ノード又は入出力端子に仮定された故障とレイアウト上現実には故障が発生し易い個所との対応を明確にすることができる。重みが付された第 1 の (未) 検出故障リスト 30a は、第 1 の (未) 検出故障リスト (重み付き) 30b に相当する。

#### 【0062】

(二) 次に、S04 段階において、少なくとも「未検出」故障を含む第 1 の (未) 検出故障リスト (重み付き) 30b から、論理回路のレイアウトに関する重みを基準にして、一部の「未検出」故障を抽出する。具体的には、レイアウトに関する重みに対応 (基本的には比例) した確率で、一部の「未検出」故障を無作為抽出する。抽出された「未検出」故障のリストは、第 2 の未検出故障リスト 31 に相当する。

#### 【0063】

(ホ) 次に、S05 段階において、非選択テストパターン 34 を用いて、第 2 の未検出故障リスト 31 の故障を検出できるか否かを計算する。即ち、第 2 の未検出故障リスト 31 に対して、非選択テストパターン 34 と、ゲートレベルネット 38 と、基本セルライブラリ 6 とを用いて故障シミュレーションを実行する。非選択テストパターン 34 によって検出された故障のリストは、追加検出故障リスト 32 に相当する。故障シミュレーションは非選択テストパターン 34 の 1 本毎に実行され、追加検出故障リスト 32 は非選択テストパターン 34 の 1 本毎に作成される。

#### 【0064】

なお、第 2 の未検出故障リスト 31 の故障には重みが付されている。よって、予めレイアウト要素の重みを除去し、レイアウト要素の重みを省略した故障に対して故障シミュレーションを行う。また、故障シミュレーションの結果、出力される追加検出故障リスト 32 には再び重みが付される。

#### 【0065】

(ヘ) 次に、S06 段階において、非選択テストパターン 34 の中から、追加検出故障リスト 32 の追加検出故障に付された重みを考慮した検出率を基準にし



て、一部のテストパターンを選択する。S 0 6 段階におけるテストパターン選択は、重み付きの「未検出」故障リスト 3 1、重み付きの追加検出故障リスト 3 2 及び選択情報記憶部 7 に記憶されている選択情報を入力情報として用いる。S 0 6 段階において選択されたテストパターンは、追加選択テストパターン 3 5 に相当する。S 0 6 段階の詳細な構成は図 5 を参照して後述する。

#### 【0 0 6 6】

(ト) 最後に、S 0 7 段階において、追加選択テストパターン 3 5 を用いて、第 1 の (未) 検出故障リスト 3 0 b の故障を検出できるか否かを計算する。即ち、第 1 の (未) 検出故障リスト 3 0 b に対して、追加選択テストパターン 3 5 と、ゲートレベルネット 3 8 と、基本セルライブラリ 6 とを用いて故障シミュレーションを実行する。S 0 7 段階における故障シミュレーションは、最終的な故障の検出率を確認することを目的とする。なお、第 1 の (未) 検出故障リスト (重み付き) 3 0 b の代わりに、第 1 の (未) 検出故障リスト 3 0 a を用いて、故障シミュレーションを行っても構わない。

#### 【0 0 6 7】

S 0 7 段階によって、論理回路に対する最終的な重み付き故障検出率及び未検出故障のリストなどが得られる。「未検出故障」とは、S 0 1 段階で選択される選択テストパターン 3 3 と S 0 6 段階で追加選択された追加選択パターン 3 5 によって検出されなかった故障であり、基本的には完全な未検出故障とハイパーアクティブ故障からなり、場合によりポテンシャル検出故障及び発振故障の一部 (割合として含む) からなる。

#### 【0 0 6 8】

図 5 に示すように、図 4 の S 0 6 段階は、S 6 0 1 段階～S 6 1 2 段階から構成されている。

#### 【0 0 6 9】

外側のループ 1 2 を回すことで、選択条件に合った (高い評価値の) テストパターンの組を選択することができる。また、内側のループ 1 3 を回すことで、適当な条件を満たしたテストパターンの中で類似した故障を検出しているものが重複して選択されるのを防ぐことができる。

**【 0 0 7 0 】**

(A) まず、S 0 5 段階が終了した後、S 6 0 1 段階において、非選択テストパターン 3 4 の 1 本毎に追加検出故障数（またはその重み）を評価する。具体的には、非選択テストパターン 3 4 の 1 本毎の追加検出故障リスト 3 2 と、評価対象リスト A 0 とを比較して、各非選択テストパターン 3 4 が評価対象リスト A 0 内の未検出故障（またはそのレイアウト要素）をどの程度検出したかを算出する。なお、最初のループ 1 2 において、「評価対象リスト A 0」は第 2 の「未検出」故障リスト 3 1 である。従って、最初のループ 1 2 においては、上記の比較をする必要はなく、各追加検出故障リスト 3 2 の追加検出故障数（またはその重み）をそのまま用いることも可能である。

**【 0 0 7 1 】**

(B) 次に、S 6 0 2 段階において、追加検出故障数（またはその重み）が 0（または 0 に近い値でも良い、以下の説明では省略）である、即ち故障を全く（または殆ど）検出していない非選択テストパターン 3 4 を、S 0 6 段階におけるテストパターンの選択対象から除外する。

**【 0 0 7 2 】**

(C) 次に、S 6 0 3 段階において、非選択テストパターン 3 4 の中に、追加検出故障数（またはその重み）が 0 でないテストパターンがあるか否かを判断する。総ての非選択テストパターン 3 4 について、追加検出故障数（またはその重み）が 0 である場合（S 6 0 3 段階において N o）、S 0 6 段階は終了し、S 0 7 段階へ進む。非選択テストパターン 3 4 の中で、追加検出故障数（またはその重み）が 0 でないテストパターンがある場合（S 6 0 3 段階において Y e s）、S 6 0 4 段階へ進む。なお、S 6 0 3 段階は、主として 2 回目以降のループ 1 2 において意味を成し、S 6 1 1 段階と共に S 0 6 段階を終了させるか否かを判断する為の段階である。

**【 0 0 7 3 】**

(D) 次に、S 6 0 4 段階において、(2) 式に示す評価値 (E V) の条件式を満たす非選択テストパターン 3 4 を候補テストパターンとして選択する。具体的には、まず、非選択テストパターン 3 4 の 1 本毎の追加検出故障リスト 3 2（

重み付き)、選択情報及び評価対象リスト A 0 (最初は「未検出」故障リスト 3 1) を元に、非選択テストパターン 3 4 の 1 本毎に評価値 (E V) を計算する。そして、その評価値 (E V) が一定の値 ( $\alpha$ ) 以上である 1 又は 2 以上のテストパターンを候補テストパターンとして選択する。なお、評価値 (E V) は、例えば、(3) 乃至 (5) 式によって算出される。

#### 【0 0 7 4】

(E) 次に、S 6 0 5 段階において、評価対象リスト A 0 を評価対象リスト B 0 へ代入する。なお、最初のループ 1 2 においては、評価対象リスト B 0 に、第 2 の「未検出」故障リスト 3 1 (評価対象リスト A 0) が代入される。S 6 0 4 段階と S 6 0 5 段階は何れが先になっても差し支えない。

#### 【0 0 7 5】

(F) 次に、S 6 0 6 段階において、対象となる候補テストパターンの評価値 (E V) を算出する。最初のステップでは評価値 (E V) は算出済みのため、改めて算出する必要はない。なお、ここでの評価値 (E V) は、最初のベストパターンによって規格化された (最初のベストパターンのデータを分母にした) ものをを用いる。

#### 【0 0 7 6】

(G) 次に、S 6 0 7 段階において、各候補パターンの追加検出故障リスト 3 2 と、評価対象リスト B 0 とを比較する。具体的には、評価対象リスト B 0 の「未検出」故障 (及びレイアウト要素) のうち、各候補パターンによってどの故障 (及びレイアウト要素) が検出され、またどの故障 (及びレイアウト要素) が未検出のままかを求める。評価対象リスト B 0 に対して各候補パターンによって追加検出される故障に伴う重みが予め決めておいた基準に達しているか否かを判断する。例えば、(6) 式の条件に従って判断する。

#### 【0 0 7 7】

候補パターンの評価値 (E V) が (6) 式の条件を満たさない候補パターンについては (S 6 0 7 段階において No)、S 6 0 9 段階に進む。そして、その候補パターンを、内側のループ 1 3 における選択対象から除外する。S 6 0 9 段階において除外される候補パターンは、候補テストパターンとして選択された時 (

S 6 0 4 段階) には追加検出故障に伴う重みが大きかったが、既に選択されているテストパターンと似通った故障を検出している。したがって、S 6 0 7 及び S 6 0 9 段階は、重複するベストパターンを除外し、冗長なテストパターンの組を形成されることを防ぐ。

#### 【 0 0 7 8 】

候補パターンの評価値 (E V) が (6) 式の条件を満たす候補パターンについては (S 6 0 7 段階において Y e s) 、S 6 0 7 . 5 段階に進む。そして、評価値 (E V) が最大の候補パターンをベストパターンとして選択確定する。(G) 次、評価対象リスト B 0 からベストパターンの追加検出故障リスト 3 2 に含まれる故障 (及びレイアウト要素) を削除して、これを新たな評価対象リスト B 0 と定義する。なお、S 6 0 7 段階において N o の候補パターンについては、評価対象リスト B 0 は変更されない。

#### 【 0 0 7 9 】

(H) 次に、S 6 1 0 段階において、総ての候補テストパターンの評価が終了したか否かを判断する。評価が終了していない場合 (S 6 1 0 段階において N o) 、S 6 0 6 段階に戻り、総ての候補テストパターンの評価が終了するまで、内側のループ 1 3 を繰り返す。内側のループ 1 3 は、効率的に未検出故障を検出している幾つかのテストパターンをベストパターンとして選び出し、追加選択テストパターン 3 5 として選択されるテストパターンの数が不用意に増加するのを防ぐ。なお、内側のループ 1 3 は、例えば、テストパターンの追加選択に比して故障検出率 (またはレイアウトの重み) の向上が殆ど見込めなくなるまで、適当な回数繰り返すことが望ましい。

#### 【 0 0 8 0 】

(I) 最後に、S 6 1 1 段階において、非選択テストパターン 3 4 の中に、追加検出故障数 (またはこれに伴うレイアウト重み) が 0 (または殆ど 0) でないテストパターンが残っているか否かを判断する。総ての非選択テストパターン 3 4 について、追加検出故障数 (またはこれに伴うレイアウト重み) が 0 (または殆ど 0) である場合 (S 6 1 1 段階において N o) 、S 0 6 段階は終了し、S 0 7 段階へ進む。非選択テストパターン 3 4 の中で、追加検出故障数 (またはこれ

に伴うレイアウト重み) が 0 (または殆ど 0) でないテストパターンが残っている場合 (S 6 1 1 段階において Yes)、S 6 1 2 段階へ進む。

#### 【0081】

S 6 1 2 段階において、評価対象リスト B 0 を評価対象リスト A 0 に上書きし、残っているテストパターンの数を、S 6 0 1 段階に示す「N」の値に代入する。「残っているテストパターン」とは、非選択テストパターン 3 4の中から、ベストパターンとして選択されているテストパターンと S 6 0 2 段階で除外されたテストパターンを除いたものを示す。そして、S 6 0 1 段階に戻り、S 6 0 3 段階或いは S 6 1 1 段階において“N o”が選択されるまで、外側のループ 1 2 を繰り返す。最終的に、S 0 6 段階が終了する時点において選択されているベストパターンが、追加選択テストパターン 3 5 に相当する。

#### 【0082】

なお、S 6 1 1 段階或いは S 6 0 3 段階において、予め、S 0 6 段階のパターン選択を終了する条件を設定しておくことができる。勿論、前述したように、残りの全てのテストパターンの追加検出故障が 0 となれば S 0 6 段階は終了する。しかし、それまでに相当数のテストパターンが選択されてしまう可能性もある。そこで、それよりも前に、例えば、残りの全てのテストパターンによって、対象とする L S I の故障検出率 (または重み付きの故障検出率) が所定の値 (例えば 0. 1 %) 未滿しか向上できないと判断した場合、S 0 6 段階を終了する方が現実的と考えられる。

#### 【0083】

また、テストパターン数を最少にしたい場合、S 0 4 段階において総ての故障を抽出して (100% サンプリング)、第 2 の未検出故障リスト 3 1 を作成した上で、S 0 6 段階のループ 1 3 の各ループにおいてベストパターンを 1 本ずつのみ選択すれば良い (S 6 0 7 段階の“=”に相当)。但し、この場合、CPU 時間が大幅に増加する場合がある。

#### 【0084】

図 6 は、図 1 に示したテストパターン選択装置の各構成要素を用いて、図 4 に示したテストパターン選択方法の手順を表現したものである。テストパターン記

憶部 3 内のテストパターン及び回路情報記憶部 5 内の R T L ネット 3 7 が、第 1 のテストパターン選択部 2 0 に入力される。具体的には、テストパターン 3 及び R T L ネット 3 7 は、機能検証カバレッジ評価部 2 8 に入力される。機能検証カバレッジ評価部 2 8 は、テストパターン 3 のおのおの（1 本毎）の機能検証カバレッジを出力する。選択実行部 2 9 は、全テストパターンで期待される故障の検出率に比較的近い故障検出率を達成する選択テストパターン 3 3 の集合と、残りの非選択テストパターン 3 4 とを出力する。

#### 【0085】

また、R T L ネット 3 7 と論理合成用の基本セルライブラリ 6 . 5 から、論理合成 3 9 により、ゲートレベルネット 3 8 が生成される。ゲートレベルネット 3 8 は、基本セルと、L S I の入出力端子を含む基本セルの入出力端子間の接続情報よりなっている。この際、一般に、メモリ等については、論理合成の対象外となっており、R T L 記述のままとなっているため、故障シミュレーションで動作させる為、利用する故障シミュレータで用意されている仕様に従ってモデル化する必要がある。こうしたメモリモデルは広い意味で基本セルライブラリに含まれるものであり、予め用意しておくことも可能である。なお、基本セルライブラリは、論理合成用のものと、故障シミュレーション用のものとで、基本セルの名前、入出力端子は同じだが、動作を記述する部分が異なり、共通して利用することはできない。

#### 【0086】

選択テストパターン 3 3 及び回路情報記憶部 5 内のゲートレベルネット 3 8 は、故障リスト記憶部 2 内の故障リスト、基本セルライブラリ記憶部 6 内の基本セルライブラリと共に、故障シミュレーション部 2 1 に入力される。故障シミュレーション部 2 1 は、第 1 の（未）検出故障リスト 3 0 a を出力する。この際、故障検出率の向上が所定の基準以下だったものは、選択テストパターンから除外し、非選択パターン 3 4 に加えることもできる（破線で示す）。

#### 【0087】

第 1 の（未）検出故障リスト 3 0 a の少なくとも「未検出」扱いの故障リスト部分は、レイアウト情報記憶部 4 内のレイアウト情報 3 6 と、レイアウト関連パ

ラメータ情報記憶部 4. 5 内のレイアウト関連パラメータ情報 3 6. 5 と共に、重み付け部 2 3 に入力される。重み付け部 2 3 は、第 1 の（未）検出故障リスト（重み付き）3 0 b を出力する。第 1 の未検出故障リスト（重み付き）3 0 b は故障抽出部 2 4 に入力され、故障抽出部 2 4 は第 2 の「未検出」故障リスト 3 1 を出力する。第 2 の「未検出」故障リスト 3 1 及び非選択テストパターン 3 4 は、ゲートレベルネット 3 8 及び基本セルライブラリ 6 と共に、故障シミュレーション部 2 1 に入力され、第 2 の故障シミュレーションが実施される。故障シミュレーション部 2 1 は、テストパターン毎に追加検出故障リスト 3 2 を作成及び出力する。

#### 【0088】

追加検出故障リスト 3 2 は、重み付きの「未検出」故障リスト 3 1、選択情報 7 と共に、第 2 のテストパターン選択部 2 6 に入力される。第 2 のテストパターン選択部 2 6 は、追加選択テストパターン 3 5 を出力する。追加選択テストパターン 3 5 は、第 1 の（未）検出故障リスト 3 0 b（または 3 0 a）、ゲートレベルネット 3 8 及び基本セルライブラリ 6 と共に、故障シミュレーション部 2 1 に入力される。故障シミュレーション部 2 1 は、重み付き故障検出率及び未検出故障リスト 4 1 などを出力する。直接重み付きの結果を出力できない場合は、重み付け部 2 3 を利用して重み付けを行う。

#### 【0089】

以上説明したように、実施の形態に係るテストパターン選択方法によれば、S 0 1 段階の機能検証カバレッジまたは故障の検出率等を基準にしたテストパターン選択には漏れたが、レイアウトに関する重みを考慮した故障検出率の向上に大きな寄与があり、かつパターン長の増加が少ないテストパターンを優先的に選択することができる。

#### 【0090】

上述したテストパターン選択方法は、時系列的につながった一連の処理又は操作、即ち「手順」として表現することができる。従って、テストパターン選択方法を、コンピュータシステムを用いて実行するために、コンピュータシステム内のプロセッサなどが果たす複数の機能を特定するコンピュータプログラムとし

て構成することができる。また、このコンピュータプログラムは、コンピュータ読み取り及び（または）書き込み可能な記録媒体または記録装置に保存することができる。この記録媒体または記録装置をコンピュータシステムによって読み込ませ、このコンピュータプログラムを実行してコンピュータを制御しながら上述したテストパターン選択方法を実現することができる。この記録媒体または記録装置は、図1に示したプログラム記憶部12を始めとする各記憶部として用いる、あるいはプログラム記憶部12を始めとする各記憶部に読み込ませ、このコンピュータプログラムにより演算部1における種々の作業を所定の処理手順に従って実行することができる。ここで、記録媒体としては、メモリ装置、磁気ディスク装置、光ディスク装置、その他のプログラムを記録することができるような装置が含まれる。

#### 【0091】

図7に示すように、テストパターン選択装置90の本体全面には、フレキシブルディスクドライブ91、及びCD-ROMドライブ92が設けられており、磁気ディスクとしてのフレキシブルディスク93または光ディスクとしてのCD-ROM94を各ドライブ入り口から挿入し、所定の読み出し操作を行うことにより、これらの記録媒体に格納されたプログラムをシステム内にインストールすることができる。また、所定のドライブ装置97を接続することにより、例えばゲームパックなどに使用されている半導体メモリとしてのROM95や、磁気テープとしてのカセットテープ96を用いることもできる。

#### 【0092】

以上説明したように、本発明の実施の形態によれば、製造品質との相関の強いレイアウト情報を反映した故障検出率を活用してテストパターンを効果的な削減することができ、与えられたテストパターンの中で製造テストの品質に最も効果のあるものだけを優先的に選択できるようになる。したがって、従来よりも少ないテストパターンで同じテスト品質を得ることができ、テストコストを削減することができる。

#### 【0093】

（その他の実施の形態）



上記のように、本発明は、1つの実施の形態によって記載したが、この開示の一部をなす論述及び図面はこの発明を限定するものであると理解すべきではない。この開示から当業者には様々な代替実施の形態、実施例及び運用技術が明らかとなろう。

#### 【0094】

実施の形態では、故障抽出部24は、レイアウトに関する重みに比例した確率で、一部の未検出故障を無作為抽出していた。しかし、本発明はこれに限定されるものではない。故障抽出部24は、未検出故障をレイアウトに関する重み順に降順で並べ替え、重みの大きい方から(7)式を用いて、第2の未検出故障リスト31の故障として抽出しても構わない。この場合、レイアウトに関する重みの大きい故障は総て考慮されることになり、より信頼性の高いテストパターン選択が実施できる。

#### 【0095】

$$X = (\text{重みの大きい故障による重みの和}) / (\text{全故障の重み}) \quad \dots (7)$$

また、第1のテストパターン選択部20は、機能検証カバレッジ28と、選択実行部29とを具備する場合について示したが、本発明はこれに限定されるものではない。例えば、第1のテストパターン選択部20は、予め対象LSIの故障をランダムにサンプリングし、これに対して各検証パターンの故障シミュレーションを実行し、故障検出率向上の寄与の大きいものから選択するようにしても良い。この際、既に第2のテストパターン選択部26において説明したような、似通ったテストパターンを重視して選択しないようなループを設ける必要がある。また、極端な場合、十分経験を積んだ設計者または検証者が、検証パターンが何をどの程度検証しているかによってテストパターンを選択するようにしても良い。

#### 【0096】

なお、本発明の別の実施形態としては、第1の故障シミュレーションを、選択テストパターン1本毎に実行し、その結果の検出故障リストを第2のテストパターン選択部26に入力し、より削減された選択テストパターンと、第1の未検出故障リスト30aまたは30b(重み付き、重み付け部23を利用)を出力する

というものがある（図面は省略）。これにより、最初の選択でも最小サイズのテストパターンが選択されることになり、全体として選択されるテストパターンのサイズがより小さくなるという効果が期待されることとなる。

#### 【0 0 9 7】

このように、本発明はここでは記載していない様々な実施の形態等を包含するということを理解すべきである。したがって、本発明はこの開示から妥当な特許請求の範囲に係る発明特定事項によってのみ限定されるものである。

#### 【0 0 9 8】

##### 【発明の効果】

以上説明したように、本発明によれば、少ないテストパターンで一定のテスト品質を得ることができ、テストコストを削減することができるテストパターン選択装置、テストパターン選択方法、及びテストパターン選択プログラムを提供することができる。

##### 【図面の簡単な説明】

#### 【図 1】

本発明の実施の形態に係るテストパターン選択装置の全体構成を示すブロック図である。

#### 【図 2】

具体的なレイアウトとレイアウト要素の例を示す図である。

#### 【図 3】

第 2 のテストパターン選択部の構成を示すブロック図である。

#### 【図 4】

本発明の実施の形態に係るテストパターン選択方法を示すフローチャートである。

#### 【図 5】

図 3 の S 0 6 段階の詳細な構成を示すフローチャートである。

#### 【図 6】

図 3 に示したテストパターン選択方法の手順を、図 1 に示したテストパターン選択装置の構成要素を用いて示したデータフロー図である。

## 【図 7】

本発明の実施の形態に係るテストパターン選択方法を実現する為のコンピュータシステムからなるテストパターン選択装置の一例を示す外観図である。

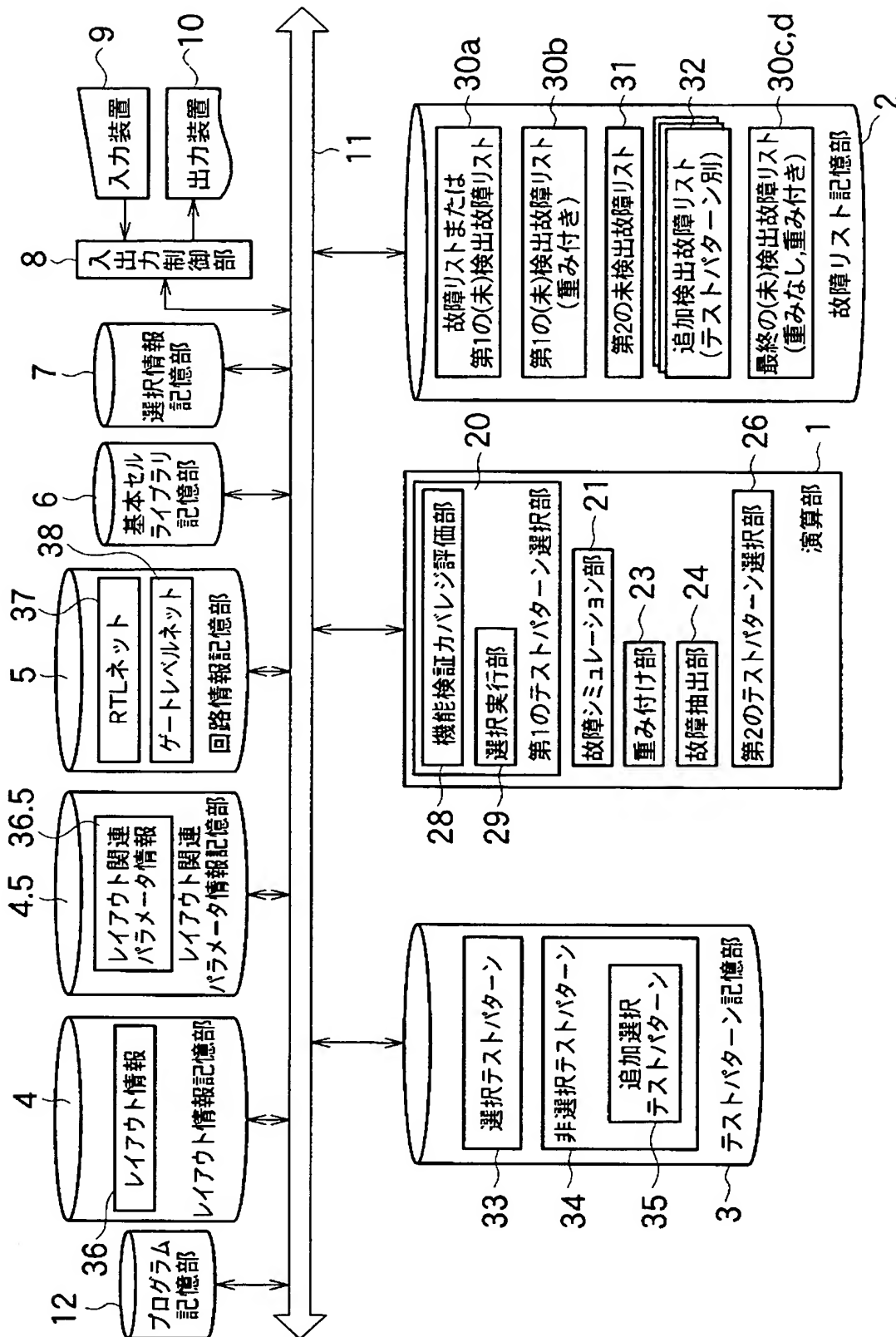
## 【符号の説明】

- 1 演算部
- 2 故障リスト記憶部
- 3 テストパターン記憶部
- 4 レイアウト情報記憶部
- 4. 5 レイアウト関連パラメータ情報記憶部
- 5 回路情報記憶部
- 6 基本セルライブラリ記憶部
- 7 選択情報記憶部
- 8 入出力制御部
- 9 入力装置
- 1 0 出力装置
- 1 1 バス
- 1 2 プログラム記憶部
- 2 0 第 1 のテストパターン選択部
- 2 1 故障シミュレーション部（第 1、第 2 及び第 3 の故障シミュレーション用）
- 2 3 重み付け部
- 2 4 故障抽出部
- 2 6 第 2 のテストパターン選択部
- 2 8 機能検証カバレッジ評価部
- 2 9 選択実行部
- 3 0 a 故障リストまたは第 1 の（未）検出故障リスト
- 3 0 b 第 1 の（未）検出故障リスト（重み付き）
- 3 0 c 最終の（未）検出故障リスト
- 3 0 d 最終の（未）検出故障リスト

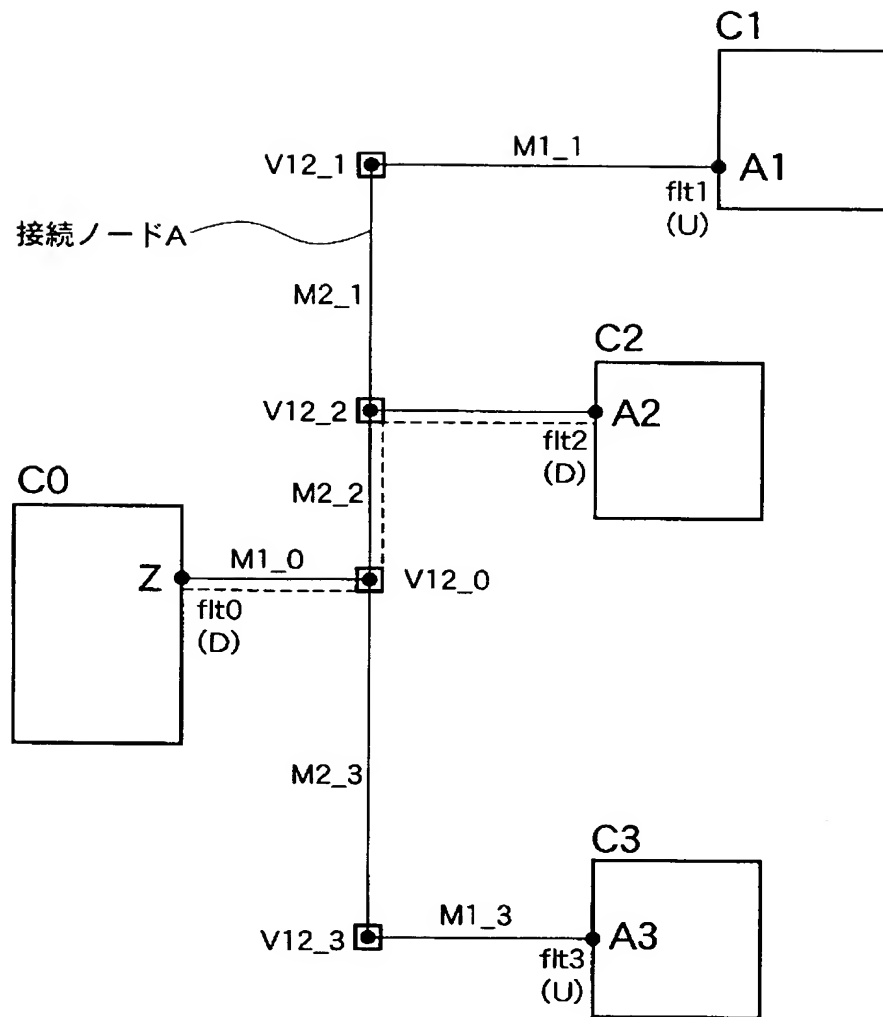
- 3 1 第 2 の未検出故障リスト
- 3 2 追加検出故障リスト（テストパターン別）
- 3 3 選択テストパターン
- 3 4 非選択テストパターン
- 3 5 追加選択テストパターン
- 3 6 レイアウト情報
- 3 6 . 5 レイアウト関連パラメータ情報
- 4 1 重み付き故障検出率／（未検出）故障リスト
- 5 1 追加検出故障数評価部
- 5 2 候補テストパターン選択部
- 5 3 ベストパターン選択部
- 5 5 評価対象リスト変更部

【書類名】 図面

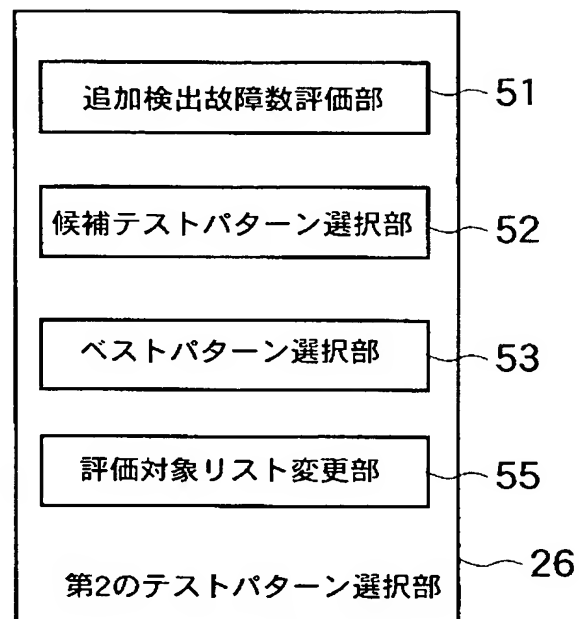
【図 1】



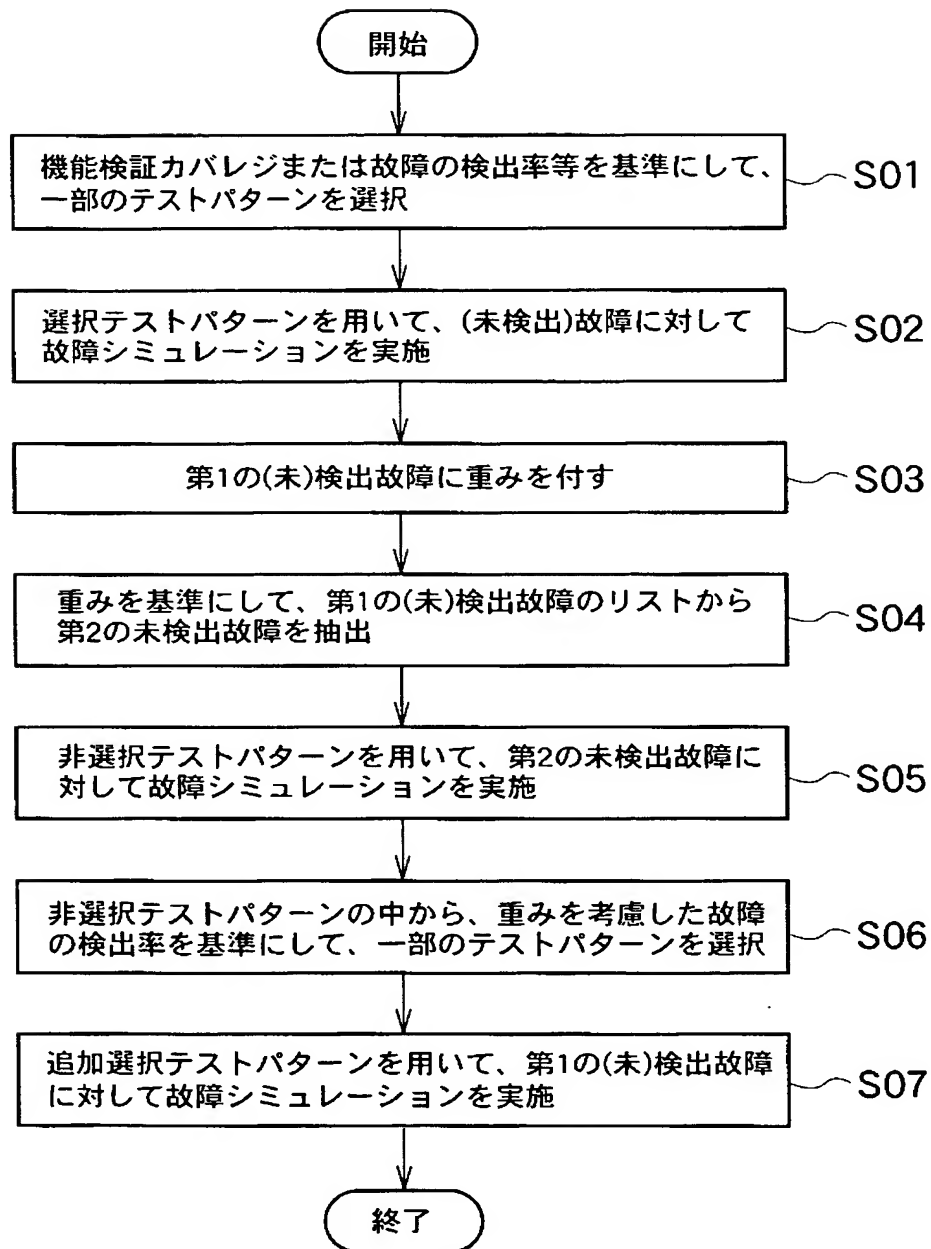
【図 2】



【図 3】

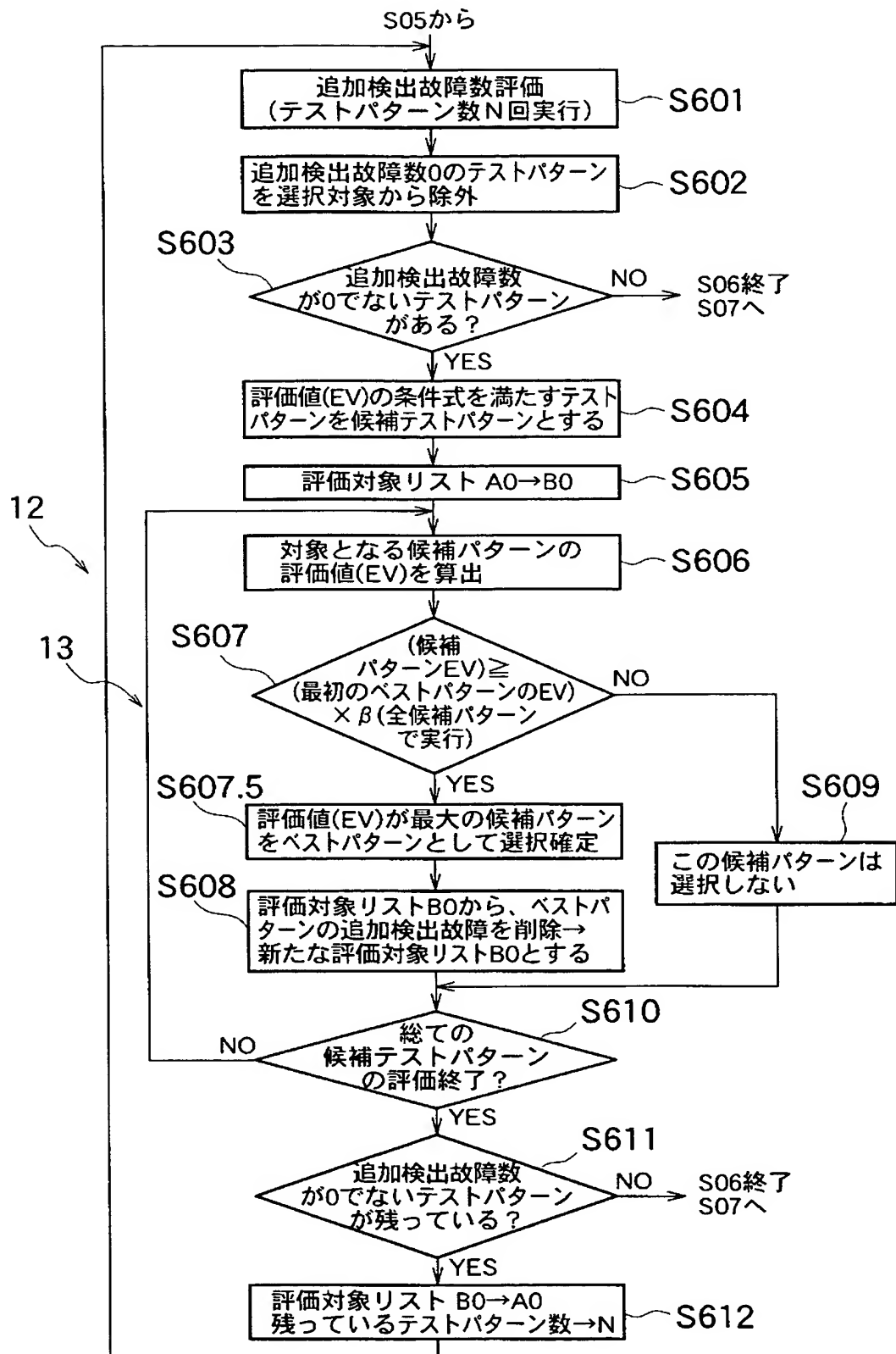


【図 4】

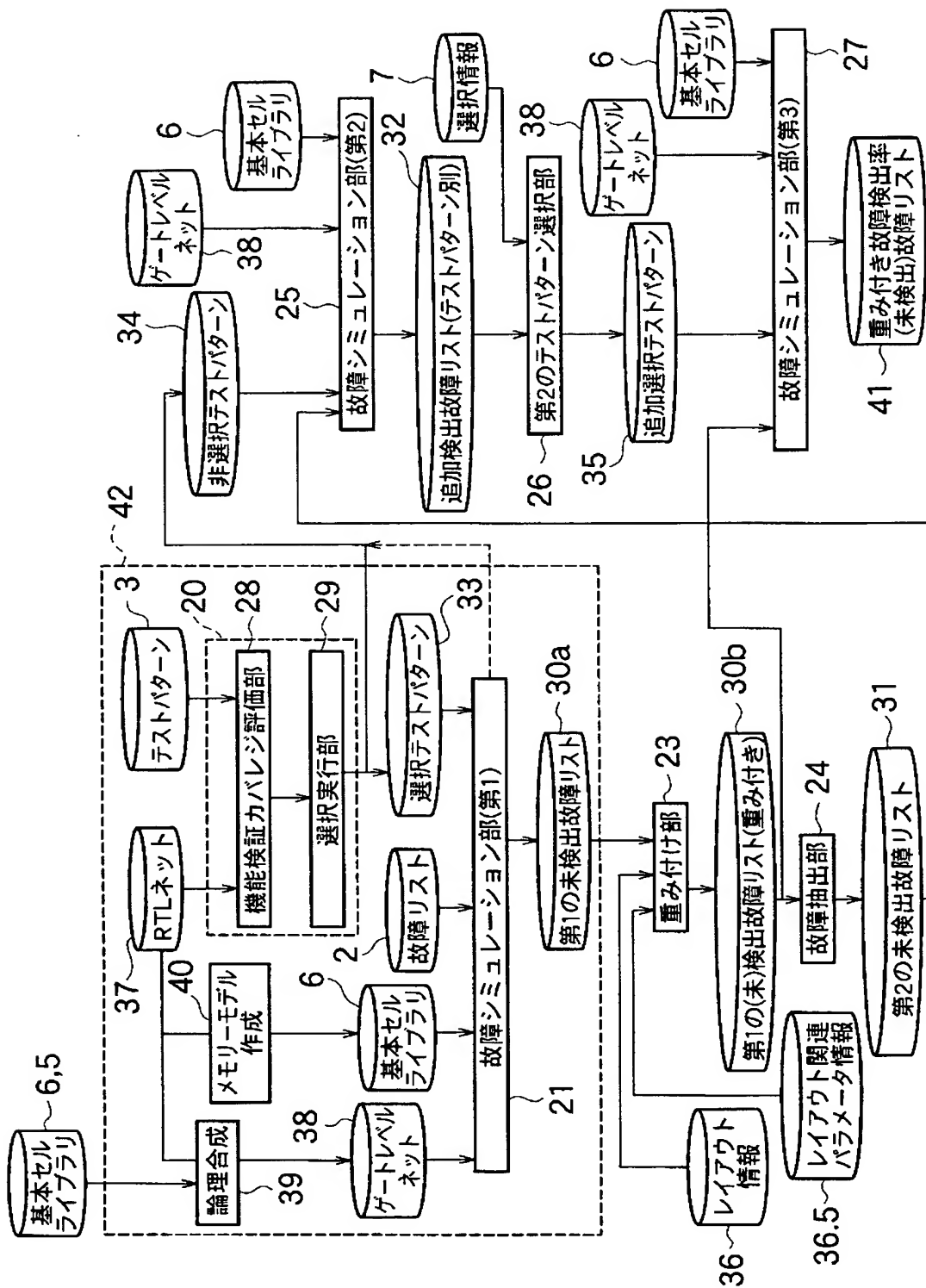




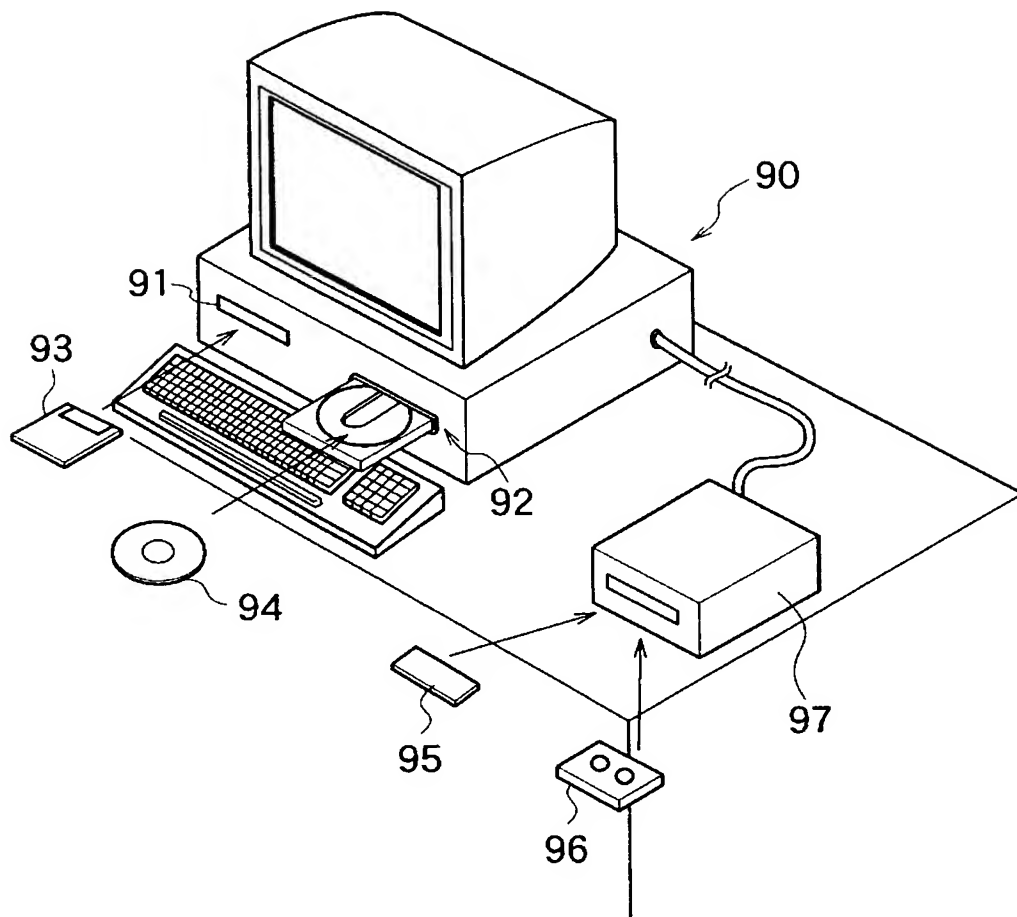
【図 5】



【図 6】



【図 7】



【書類名】 要約書

【要約】

【課題】 少ないテストパターンで一定のテスト品質を得ることができ、テストコストを削減できるテストパターン選択装置を提供する。

【解決手段】 論理回路内に仮定される故障を検出する為に、検証パターンの集合から機能検証カバレッジまたは故障の検出率を基準にして一部の検証パターンを第 1 のテストパターンとして選択する第 1 のテストパターン選択部と、選択された第 1 のテストパターン（選択テストパターン）によって検出されない故障（第 1 の未検出故障）にレイアウトに関する重みを付す重み付け部と、選択されなかった検証パターン（非選択検証パターン）の中から第 1 の未検出故障に付された重みを考慮して一部の検証パターンを第 2 のテストパターンとして選択する第 2 のテストパターン選択部とを有する。

【選択図】 図 1

特願 2 0 0 2 - 2 9 1 4 4 2

出 願 人 履 歴 情 報

識別番号

[ 0 0 0 0 0 3 0 7 8 ]

1 . 変更年月日

2 0 0 1 年    7 月    2 日

[変更理由]

住所変更

住 所

東京都港区芝浦一丁目 1 番 1 号

氏 名

株式会社東芝